Listing of Claims

Claim 1 (Currently Amended): A method of enabling a user to ~~customize~~ extend a work flow ~~associated with an operation~~ for synchronization/consolidation of data between at least two data sources, said work flow for execution in a meta directory server, ~~said operation requiring communication with at least two data sources,~~ said method comprising:

providing a plurality of built-in tasks which together when executed implement said work flow ~~to implement said operation requiring communication with said two data sources~~, at least one of said plurality of built-in tasks containing an extension point;

receiving from said user data indicating a custom task associated with said extension point, wherein said custom task is separate from said plurality of built-in tasks and contains a program logic specified by said user; and

executing said custom task when said extension point is reached during execution of said one of said plurality of built-in tasks.

2 (Currently Amended): The method of claim 1, wherein said plurality of built-in tasks are provided by a designer implementing said meta directory server such that another program logic constituting said plurality of built-in tasks cannot be edited by said user,

_____wherein said designer is different from said user,

_____ whereby each user can provide custom extensions to said work flow by providing a corresponding instance of said program logic for said custom task.

Claim 3 (Original): The method of claim 1, wherein said custom task contains an another extension point, said method further comprises receiving from said user data

indicating an another custom task to be executed when said another extension point is reached during execution of said custom task.

Claim 4 (Original): The method of claim 3, further comprising:

determining a corresponding set of extension points available in each of said plurality of built-in tasks;

displaying each of said set of extension points associated with a corresponding one of said plurality of built-in tasks;

displaying said custom task and said another custom task; and

enabling said user to specify said custom task associated with said extension point, and said another custom task associated with said another extension point.

Claim 5 (Original): The method of claim 3, further comprising enabling said user to specify that said custom task is to be executed synchronously, wherein said custom task is executed in a synchronous manner.

Claim 6 (Original): The method of claim 3, further comprising enabling said user to specify that said custom task is to be executed asynchronously, wherein said custom task is executed in a asynchronous manner.

Claim 7 (Canceled)

Claim 8 (Currently Amended): The method of claim 1 7, wherein at least one of said two data sources comprises a relational database.

Claim 9 (Original): The method of claim 3, further comprising providing an utility to indicate that a specific one of said extension points is reached.

Claim 10 (Original): The method of claim 3, further comprising providing an utility in each of said plurality of built-in tasks and said custom task, wherein said utility indicates extension points available in a corresponding task.

Claim 11 (Currently Amended): A computer readable medium storing ~~carrying~~ one or more sequences of instructions for causing a meta directory server to enable a user to ~~customize~~ extend a work flow ~~associated with an operation~~ for synchronization/consolidation of data between at least two data sources, ~~said operation requiring communication with at least two data sources,~~ wherein execution of said one or more sequences of instructions by one or more processors contained in said meta directory server causes said meta directory server ~~one or more processors~~ to perform the actions of:

providing a plurality of built-in tasks which together when executed implement said work flow ~~to implement said operation requiring communication with said two data sources~~, at least one of said plurality of built-in tasks containing an extension point;

receiving from said user data indicating a custom task associated with said extension point, wherein said custom task is separate from said plurality of built-in tasks and contains a program logic specified by said user; and

executing said custom task when said extension point is reached during execution of said one of said plurality of built-in tasks.

Claim 12 (Currently Amended): The meta directory server of claim 11, wherein said plurality of built-in tasks are provided by a designer implementing said meta directory server <u>such that another program logic constituting said plurality of built-in tasks cannot be edited by said user,</u>

_____wherein said designer is different from said user<u>,</u>

<u>_____ whereby each user can provide custom extensions to said work flow by providing a corresponding instance of said program logic for said custom task.</u>

Claim 13 (Original): The meta directory server of claim 11, wherein said custom task contains an another extension point, further comprises receiving from said user data indicating an another custom task to be executed when said another extension point is reached during execution of said custom task.

Claim 14 (Original): The meta directory server of claim 13, further comprising:

determining a corresponding set of extension points available in each of said plurality of built-in tasks;

displaying each of said set of extension points associated with a corresponding one of said plurality of built-in tasks;

displaying said custom task and said another custom task; and

enabling said user to specify said custom task associated with said extension point, and said another custom task associated with said another extension point.

Claim 15 (Original): The meta directory server of claim 13, further comprising enabling said user to specify that said custom task is to be executed synchronously, wherein

said custom task is executed in a synchronous manner.

Claim 16 (Original): The meta directory server of claim 13, further comprising

enabling said user to specify that said custom task is to be executed asynchronously, wherein

said custom task is executed in a asynchronous manner.

Claim 17 (Canceled)

Claim 18 (Currently Amended): The meta directory server of claim 11 ~~17~~, wherein

at least one of said two data sources comprises a relational database.

Claim 19 (Original): The meta directory server of claim 13, further comprising

providing an utility to indicate that a specific one of said extension points is reached.

Claim 20 (Original): The meta directory server of claim 13, further comprising

providing an utility in each of said plurality of built-in tasks and said custom task, wherein

said utility indicates extension points available in a corresponding task.

Claim 21 (Currently Amended): A meta directory server enabling a user to ~~customize~~

extend a work flow ~~associated with an operation~~ for synchronization/consolidation of data

between at least two data sources, ~~said operation requiring communication with at least two~~

~~data sources,~~ said meta directory server comprising:

means for providing a plurality of built-in tasks which together when executed

implement said work flow ~~to implement said operation requiring communication with said~~

~~two data sources~~, at least one of said plurality of built-in tasks containing an extension point;

means for receiving from said user data indicating a custom task associated with said extension point, wherein said custom task is separate from said plurality of built-in tasks and contains a program logic specified by said user; and

means for executing said custom task when said extension point is reached during execution of said one of said plurality of built-in tasks.

Claim 22 (Currently Amended):   The meta directory server of claim 21, wherein said plurality of built-in tasks are provided by a designer implementing said meta directory server such that another program logic constituting said plurality of built-in tasks cannot be edited by said user,

_____wherein said designer is different from said user,

_____whereby each user can provide custom extensions to said work flow by providing a corresponding instance of said program logic for said custom task.

Claim 23 (Original): The meta directory server of claim 21, wherein said custom task contains an another extension point, further comprises means for receiving from said user data indicating an another custom task to be executed when said another extension point is reached during execution of said custom task.

Claim 24 (Original): The meta directory server of claim 23, further comprising:

means for determining a corresponding set of extension points available in each of said plurality of built-in tasks;

means for displaying each of said set of extension points associated with a

corresponding one of said plurality of built-in tasks;

means for displaying said custom task and said another custom task; and

means for enabling said user to specify said custom task associated with said extension point, and said another custom task associated with said another extension point.

Claim 25 (Original): The meta directory server of claim 23, further comprising means for enabling said user to specify that said custom task is to be executed synchronously, wherein said custom task is executed in a synchronous manner.

Claim 26 (Original): The meta directory server of claim 23, further comprising means for enabling said user to specify that said custom task is to be executed asynchronously, wherein said custom task is executed in a asynchronous manner.

Claim 27 (Canceled)

Claim 28 (Currently Amended): The meta directory server of claim 21 27, wherein at least one of said two data sources comprises a relational database.

Claim 29 (Original): The meta directory server of claim 23, further comprising an utility means to indicate that a specific one of said extension points is reached.

Claim 30 (Original): The meta directory server of claim 23, further comprising an utility means in each of said plurality of built-in tasks and said custom task, wherein said utility means indicates extension points available in a corresponding task.

Claim 31 (Currently Amended): A meta directory server enabling a user to ~~customize~~ extend a work flow ~~associated with an operation~~ for synchronization/consolidation of data between at least two data sources, ~~said operation requiring communication with at least two data sources,~~ said meta directory server comprising:

a task registry block storing data related to a plurality of built-in tasks which together when executed implement said work flow ~~to implement said operation requiring communication with said two data sources~~, at least one of said plurality of built-in tasks containing an extension point;

a user interface module receiving from said user, data indicating a custom task associated with said extension point, wherein said custom task is separate from said plurality of built-in tasks and contains a program logic specified by said user; and

work-flow manager module for executing said custom task when said extension point is reached during execution of said one of said plurality of built-in tasks.

Claim 32 (Currently Amended): The meta directory server of claim 31, wherein said plurality of built-in tasks are provided by a designer implementing said meta directory server such that another program logic constituting said plurality of built-in tasks cannot be edited by said user,

_____wherein said designer is different from said user,

_____whereby each user can provide custom extensions to said work flow by providing a corresponding instance of said program logic for said custom task.

Claim 33 (Original): The meta directory server of claim 31, wherein said custom task contains an another extension point, wherein said user interface further receives data

indicating an another custom task to be executed when said another extension point is reached during execution of said custom task.

Claim 34 (Original): The meta directory server of claim 33, wherein said user interface modules displays each of said set of extension points associated with a corresponding one of said plurality of built-in tasks, and enables said user to specify said custom task associated with said extension point and said another custom task associated with said another extension point.

Claim 35 (Original): The meta directory server of claim 33, wherein said user interface enables said user to specify that said custom task is to be executed synchronously, wherein said custom task is executed in a synchronous manner.

Claim 36 (Original): The meta directory server of claim 33, wherein said user interface enables said user to specify that said custom task is to be executed asynchronously, wherein said custom task is executed in a asynchronous manner.